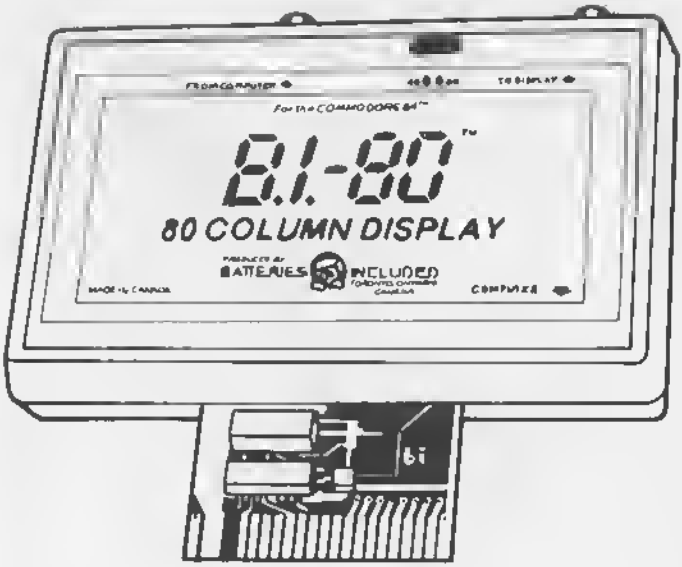


BATTERIES INCLUDED

TORONTO, ONT. CANADA

B I - 8 0

80 Column Display

by Batteries Included

Thank you for purchasing BI-80.

We believe that BI-80 is the best 80 column display available for the Commodore 64. BI-80 provides a crystal clear 80 column display and powerful Basic 4.0 language extension. BI-80 installs easily and is compatible with many programs including the PaperClip64 wordprocessor and the Consultant database.

Again, Thank you for choosing BI-80.

Batteries Included.

Batteries Included
186 Queen Street West
Toronto, Ontario
Canada, M5V-1Z1
(416) 596-1405

Copyright 1984 by Batteries Included

All Rights Reserved

Programming: Steven Douglas
Hardware design: PME
Produced by Batteries Included

Second Edition

NOTICE OF REGISTERED TRADEMARK

Reference is made in several places in this manual to Commodore and Commodore 64 which are registered trademarks of Commodore Business Machines Inc.

BI-80 OWNERS MANUAL

COPYRIGHT 1984 BY BATTERIES INCLUDED

186 Queen Street West
TORONTO, ONTARIO CANADA, M5V 1Z1
(416) 596-1405

This manual and the computer programs contained in or transferred from the BI-80 ROM chip which are described by this manual are copyrighted and contain proprietary information belonging to Batteries Included.

This manual may not be copied, photocopied, reproduced, translated or reduced to machine readable form, in whole or in part, without the prior written consent of Batteries Included.

The ROM chip, and information contained within it, may not be duplicated, in whole or in part, for any purpose. No copies of this manual or the listings of the programs in the ROM chip may be sold or given to any person or other entity.

LIMITATIONS OF WARRANTY AND LIABILITY

Batteries Included, or any dealer or distributor distributing this product, makes NO WARRANTY, EXPRESS OR IMPLIED, with respect to this manual, the BI-80 software, hardware and any other related items, their quality, performance, merchantability, or fitness for any particular use. It is solely the purchaser's responsibility to determine their suitability for any particular purpose.

Batteries Included will in no event be held liable for direct, indirect or incidental damages resulting from any defect or omission in this manual, the BI-80 ROM, the BI-80 hardware, the software contained within the ROM chip, or other related items and processes, including but not limited to any interruption of services, loss of business or anticipatory profit, or other consequential damages.

THIS STATEMENT OF LIMITED LIABILITY IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Batteries Included neither assumes nor authorizes any other person to assume for it any other warranty or liability in connection with the sale of its products.

PRODUCT IMPROVEMENTS

Batteries Included reserves the right to make corrections or improvements to this manual and to the related BI-80 device at any time without notice and with no responsibility to provide these changes to purchasers of earlier versions of its products.

Unpacking

Before installing the BI-80 in your Commodore 64 check the package you received to make sure you have everything listed below.

- BI-80 80 Column display adapter

- BI-80 cables (male phono to male phono)
  (male phono to female phono)

- Warranty registration card

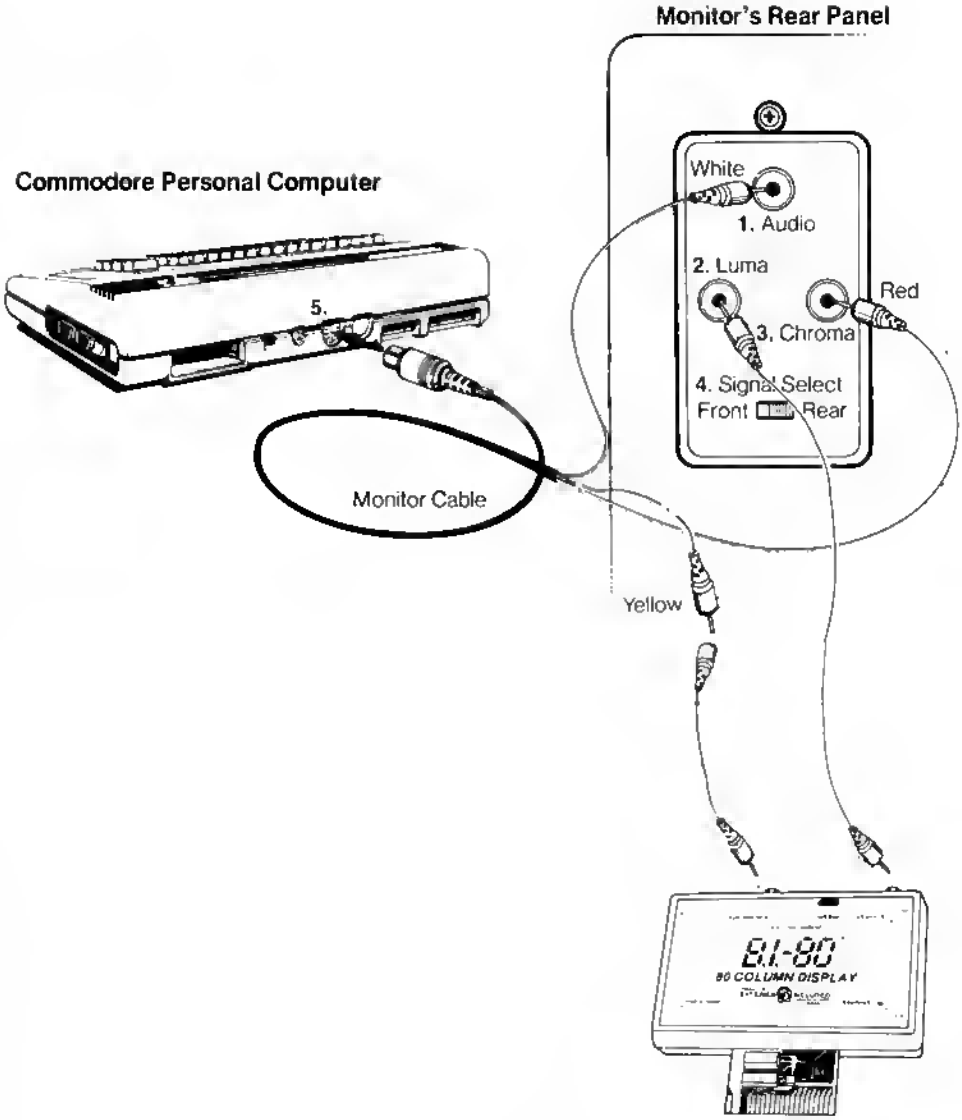
In addition, you should have on hand the following:

- Commodore monitor or any good color or monochrome monitor.

Installation

Make sure that the power for your Commodore 64 computer is disconnected and that any cartridges are removed. Insert the BI-80 into the cartridge slot of your computer. Make sure it is fully seated in the connector. Turn the computer on again. It should power up and display the sign-on message as in Figure 1. Turn the unit off.

Remove the cable from the luminance connection on the back of your monitor. Connect the male phono to male phono cable from the connector on the right rear corner of the BI-80 (to monitor) to the luminance connection on your monitor. Connect the female end of the male to female phono cable to the cable you unplugged from your monitor. Connect the male end to the left rear connection on the BI-80 (from computer). Refer to Figure 2.


Monitor's Rear Panel
Commodore Personal Computer
White
1. Audio
2. Luma
Red
3. Chroma
4. Signal Select
Front
Rear
5.
Monitor Cable
Yellow
80 COLUMN DISPLAY

General Configuration

BI-80 transforms your Commodore 64 into a powerful 80 column computer. Your screen can now display a crystal clear 80 columns by 25 lines and Basic 4.0 provides powerful disk commands with both 40 and 80 column displays.

When the Commodore 64 is first turned on with the BI-80 connected, the 40 column display is selected. The Basic 4.0 language extension is active. To enable the 80 column display enter the command **SYS 33000**. The 80 column display will appear. To return to the 40 column system enter the command **SYS 33003**.

BI-80 reduces the amount of memory available for the Basic language by 8K. If you have to remove the BI-80 to allow a program to operate you will have to unplug the video cables.

Basic 4.0

Basic 4.0 is a set of additional commands added to the regular set available in the standard Basic 2.0 language. This section is intended to outline and explain in simple terms the additional commands available with the Basic 4.0 language. It is not intended to teach programming or explain in detail the techniques which can utilize these commands. For more comprehensive information the following sources may be useful:

User's Reference Manual - Commodore BASIC Version 4.0
Part# 321604 - Commodore Business Machines Inc.

User's Manual for CBM 5 1/4-inch Dual Floppy Disk Drives
Part# 320899 - Commodore Business Machines Inc.

PET PERSONAL COMPUTER GUIDE
- Adam Osbourne, Jim Strasma, Ellen Strasma
Copyright 1982 - Osbourne/McGRaw-Hill - ISBN 0-931988-76-4

The following convention has been used for the Basic 4.0 command descriptions. Commands are in UPPER case, variable data is in lower case. Square brackets [and] indicate an optional parameter. Quotes "" indicate that a string constant or variable is required, Angle brackets <> indicate that a constant or variable may be used. In most cases variables must be enclosed in brackets () to avoid a SYNTAX ERROR.

Turning Basic 4.0 ON and OFF

When the computer is first powered on the Basic 4.0 language extension is enabled. To disable the Basic 4.0 language and return operation to the standard Basic 2.0 language enter the command **SYS 33009** . Enable it with **SYS 33006** . Do NOT enable or disable the Basic 4.0 language from within a Basic program - only from the keyboard in immediate mode.

DS and DS$ (Disk Status)

Because the disk drive controller is a separate computer, any errors it encounters during operation must be manually retrieved from the drive to the computer. Error messages are usually indicated by a RED centre light on dual slot drives, or a rapidly flashing RED light on single slot drives. Basic 4.0 can retrieve and display an error message from the disk drive simply by referring to the pseudo-variables DS and DS$. DS and DS$ are used as if they were regular Basic variables, except that you cannot assign values to them. Instead, when you refer to either DS or DS$ after issuing a disk related Basic 4.0 command (such as DSAVE), the computer will fetch the current error status of the disk unit last referred to and store the resulting string in DS$. The first number in the DS$ string is stored in DS - this is known as the error code. It is important to note that DS and DS$ are only updated when you refer to them AND a disk related command has been issued since the last reference to DS or DS$. If you refer to DS or DS$ and a disk command has not been issued since the last reference, then the current values are reused and the drive is not queried. This allows programming of the form:
IF DS <> 0 THEN PRINT DS$: STOP

There are three messages which are not errors (and do not light the error lamp) which may be retrieved by DS and DS$.

73, cbm dos v.....,00,00 - This message waits in the drive after power-up until a command is executed.

00, ok,00,00 - This is the message returned if the previous command was executed with no errors. It is also returned when there was an error, DS or DS$ was referenced, and for some reason the drive was queried a second time without any disk command being issued (this normally shouldn't happen since DS and DS$ keep track of disk commands).

01, files scratched,xx,00 - This message is only returned after a SCRATCH command is executed by the drive. The two digits 'xx' indicate the number of files actually scratched by the drive - it will be zero if no files were scratched.

Note 1: If no device responds to Basic 4.0's request for the disk message then DS will have a value of 0 to 20. DS$ will have a length of zero.

Note 2: The act of fetching DS or DS$ will cause ST to be changed - therefore you must save ST before checking DS and DS$ if ST is needed later in the program.

STATUS - (ST) - result of last I/O operation

ST is used as if it was a regular Basic variable, except that you cannot assign a value to it. It is used as a variable which contains the result of the last attempted I/O operation. The value will vary depending on the success of the I/O with the various bits indicating exact nature of the problem.

Bit position	ST numeric value	IEEE meaning
0	1	time out on write
1	2	time out on read
2	4	none
3	8	none
4	16	none
5	32	none
6	64	EOI (end or identify)
7	-128	device not present

Status values of 1 and 2 indicate that the device you are trying to communicate with is not accepting or sending data. For instance. If when reading from a sequential file the program attempts to read more data than exists in the file, the drive will TIME OUT ON READ and status will equal 2. A status value of 64 on reading from a file indictes that the last data item fetched is the last data item in the file and there are no more to follow The computer often generates a status of 64 when it is sending date to the disk drive or printer. If status is -128, then the device number you are trying to communicate with does not exist.

DOS file name pattern matching

The Commodore DOS systems allow considerable flexibility in specifying which files a command will act upon. For example. If you wished to scratch (delete) a file or group of files, you could specify the file name in many different ways:

```
fred      -the single file 'fred'
fred*     -all files starting with the characters 'fred'
fr?d      -all files matching 'fr?d'. where '?' is any character
fr?d*     -all files matching 'fr?d' with anything following
0:fred    -the file 'fred' on drive 0
1:fred*   -all files on drive 1 starting with 'fred'
fred=prg  -the program file 'fred'
0:f*=seq  -all sequential files on drive 0 which start with 'f'
1:*=rel   -all relative files on drive 1
```

Many disk commands can use some or all of the methods shown above to specify files. Others will require a specific format as described in the individual commands.

Basic 4.0 Commands

APPEND

Format - APPEND# <file number> , "<name>" [,D<x>] [ON U<y>]

Use - To add additional data to the end of a sequential disk file.

Notes - APPEND is used like a DOPEN command but can only be used to add data to an existing sequential disk file. APPEND opens the specified data file for write and positions the DOS pointers to the current end of the file and new data can be added. Any variable or evaluated expressions must be enclosed in parentheses. Unit defaults to device 8, drive 0.

Example - x=1 : APPEND#1, "data file", D(x) ON U 9
Opens a file with a logical number of 1 called "data file" on drive 1 of unit 9 for append.

BACKUP

Format - BACKUP D<x> to D<y> [ON U<z>]

Use - To create an exact duplicate of a diskette using a dual drive

Notes - Not applicable for single slot drives.
Both drive numbers must be specified and must be 0 and 1 Unit defaults to device 8. Any variable or evaluated expressions must be enclosed in parentheses.

Example - BACKUP D0 to D1

CATALOG or DIRECTORY

Format - CATALOG [“<pattern>”,] [D<x>] [ON U<y>]
DIRECTORY [“<pattern>”,] [D<x>] [ON U<y>]

Use - Displays on the current output device (usually the screen) the disk directory from the specified drive. If no drive is specified both drives are searched in a dual drive. Any variable or evaluated expressions must be enclosed in parentheses. Unit defaults to device 8. Pattern matching as described on Page 3-3 may be used to view limited segments of the directory. Pressing the SPACE bar will cause the scroll of the display to pause, pressing it again will continue the display. STOP terminates the listing.

Example - CATALOG DO DN U 9
DIRECTORY DD “fred*”

COLLECT

Format - COLLECT [D<x>] [ON U <y>]

Use - Causes the disk drive to review the block allocations for all files in the directory, deleting all incorrectly closed files (those marked with an asterisk in the directory). The block count and allocation map are updated to correspond correctly to the files currently on the disk. Space allocated by the BLOCK-ALLOCATE command is freed since it is not linked to the directory. Any variable or evaluated expressions must be enclosed in parenthesis. Unit defaults to device 8, drive D

Example - COLLECT DD

CONCAT

Format - CONCAT [D<x>,] <“namea”> TO [D<y>,] “nameb” [ON U<z>]

Use - Causes the sequential file “nameb” to have the data in “namea” appended to it. The data in the file “namea” is unaltered. Can only be used with sequential data files. Any variable or evaluated expressions must be enclosed in parentheses. Unit defaults to device 8, drive D.

Example - CONCAT “two” TO “one” ,D1
creates file “one” which contains “one” + “two”

COPY

Format - COPY [D<x>,] "<namea>" TO [D<y>,] "<nameb>" [ON U<z>]
or - COPY D<x> TO D<y> [ON U<z>]

Use - causes the disk drive to copy a file or files from one place to another within a disk unit. It can be used to copy data from one disk to another in a dual drive and to copy a file to another place on the same diskette. Pattern matching as described on Page 3-3 may be used to copy multiple files. If the file is to be copied to the same diskette then "nameb" must differ from "namea". COPY without file names copies all files from one diskette to the other in a dual drive without erasing those already there. Any variable or evaluated expressions must be enclosed in parentheses. Unit defaults to device 8.

Example - COPY 00 to 01
COPY 01 "temp" TO "tempb"

DCLOSE

Format - DCLOSE [#<x>] [ON U<y>]

Use - Close one or many currently open disk files. DCLOSE will only close those disk files which were currently open in the computer's internal file table. If a logical file number <x> is specified then only that file will be closed. If only the unit number <y> is specified then all files on that unit will be closed. If both logical and unit numbers are specified then the unit number is ignored. If no parameters are given then all disk files will be closed. Relative record files may have more records generated at close time than were actually written in order to fill out the sectors in use. Any variable or evaluated expressions must be enclosed in parentheses. Unit defaults to device 8.

Example - DCLOSE :closes all files currently open on device 8
DCLOSE#2 :close the file with logical number 2

DLOAD

Format - DLOAD "<name>" [,D<x>] [ON U<y>]

Use - Load a Basic program from disk, relocating it if necessary. Any variable or evaluated expressions must be enclosed in parentheses. Pattern matching as described on Page 3-3 may be used. If an asterisk (*) is used as the filename the last accessed program on the disk will be loaded (If you exchange diskettes after the first load then this will not work). If the last file accessed was not a program file then the first program file in the disk directory will be loaded. Unit defaults to device 8. If no drive is specified both drives are searched in a dual drive. If used in program mode the subsequently loaded program will begin execution with the first line.

Example - DLOAD "first file" ,D1

DOPEN

Format - DOPEN# <a> , "<name>" [,L<y>] [,D<x>] [ON U<z>] [,W]

Use - Open a sequential or random access file for read or write. <a> is the logical file number. Logical file numbers greater than 127 cause each PRINT# statement to transmit a LINE FEED character and a CARRIAGE RETURN character. The L<y> parameter only needs to be specified when opening a relative record file for the first time. Relative record files are always opened for both read and write. If not specified, sequential files are opened for read. Any variable or evaluated expressions must be enclosed in parentheses. Unit defaults to device 8, drive 0. Both drives in a dual drive will be searched if all the optional parameters are missing. If the first character of the filename is the @ (at) symbol it is removed and the specified file is opened replacing any existing file with that name provided that the file type is the same.

Example - DOPEN# 5 , "fred" , L 27 , D1
open a relative file. record length of 27 bytes, on drive 1

DOPEN# 2 , "@abc file", W
opens sequential file on drive 0 for write with a logical file # of 2.
If "abc file" previously existed it has now been replaced.

DSAVE

Format - DSAVE "<name>" [,D<x>] [ON U<y>]

Use - Save a Basic program to disk. The "name" parameter can be up to 16 characters long. Any variable or evaluated expressions must be enclosed in parentheses. Unit defaults to device 8, drive 0. If the first character of the filename is the @ (at) symbol, it is removed and the specified file replaces any existing file with that name provided that the file type is the same.

Example - DSAVE "my program" ,D1
DSAVE "@freds prog"
save "freds prog" on drive D replacing any previous program file named "freds prog".

HEADER

Format - HEADER "<disk name>" [,D<x>] [I<zz>] [ON U<y>]

Use - To format (new) a new disk or erase the contents of an old one. The <disk name> may be up to 16 characters long. A 'STRING TOO LONG error will occur if the <disk name> is greater than 16 characters long. If the I<zz> parameter is missing then the drive will attempt to reformat only the directory track. The <zz> parameter may be specified as a string variable. Any variable or evaluated expressions must be enclosed in parantheses. Unit defaults to device 8, drive 0.

When used in immediate mode the question 'ARE YOU SURE?' will be asked before the command proceeds. Answer Y or YES to continue, N to abort. On completion the screen will display **READY** if the disk was correctly formatted or **BAD DISK** if any error was encountered. The actual error can be displayed by printing the contents of DS$.

CAUTION: The HEADER command will erase all information previously stored on the disk! Be careful!

Example - HEADER "my first disk" , D1 ON U 8 , Ibz
HEADER "next disk"

RECORD

Format - RECORD# <logical file#> , <record#> [,<byte position>]

Use - Used before a GET#, INPUT# or PRINT# statement to position to the correct record number in a random access data file opened with the <logical file number>. <record#> must be between 1 and 65535 inclusive. <byte position> must be between 1 and 254 inclusive. <byte position> defaults to 1 if omitted. Any variable or evaluated expressions must be enclosed in parentheses.

If the <record#> causes the pointer to be positioned beyond the current end of file (as it would be when adding data to the file) the disk status (DS$) will show a RECORD NOT PRESENT error. If the intent is to add data to the file, then the next PRINT# statement will cause that record and any necessary intervening records to be generated. If INPUT# is executed then a null value is returned with STATUS (ST) set to EOI (64).

Example - RECORD# 1 402 , 5
positions to byte 5 of record 402

RENAME

Format - RENAME [D<x>,] "<old name>" TO "<new name>" [ON U<y>]

Use - Changes the name of a specified disk file. The file must not be currently open. Any variable or evaluated expressions must be enclosed in parentheses. Unit defaults to device 8, drive 0. It is not possible to RENAME a file to a name already existing on the disk.

Example - RENAME "freds file" TO "my file" , D1

SCRATCH

Format - SCRATCH "<name>" [,D<x>] [ON U<y>]

Use - Erase a disk file or files. Any variable or evaluated expressions must be enclosed in parentheses. Unit defaults to device 8, drive 0. When used in immediate mode the question 'ARE YOU SURE?' will be asked before the command proceeds. Answer Y or YES to continue, N to abort. Multiple files may be scratched at one time by using the pattern matching as described on Page 3-3

Example - SCRATCH "my file"
erase only "my file" from drive 0

SCRATCH "fred*" ,d1
erase all files on drive 1 beginning with "fred"

AUTO RUN (shift RUN/STOP)

Format - Press the RUN/STOP key while holding the SHIFT key

Use - Pressing the RUN/STOP key while holding the SHIFT key down will cause the computer to load and run the first program on drive 0 of device 8.

The BI-80 display acts as an 8K cartridge residing at $8000. On power up the Basic 4.0 is engaged, the 80 column screen initialized, cleared and the standard 40 column screen is displayed. The Basic 4.0 language is separate from the 80 column system and can be enabled and disabled without affecting the 80 column operation. The 80 column screen memory resides at $9800 to $9FFF. If the lomem control bit in the 6510 is turned off, the BI-80 ROM at $8000 is disabled, but the screen RAM is not. Any write to memory from $9800 to $9FFF will write to both the memory in the computer and in the BI-80.

Several vectors are available in the BI-80 display.

The BI-80 ROM is coded with the string **'batteries included'** starting at address 32777 ($8009).

Vector decimal	hex	Operation
33000	$80E8	- Engage 80 column operating system, initialize hardware, switch to 80 column display.
33003	$80EB	- Engage standard 40 column operating system, switch to 40 column display
33006	$80EE	- Enable BASIC 4.0 language extension. (Note: Basic 4.0 is enabled on power up).
33009	$80F1	- Disable Basic 4.0 language extension.
33012	$80F4	- Initialize 6545 CRTC chip, switch to 80 column display, (80 columns, 25 lines).
33015	$80F7	- Install 80 column operating system vectors.
33018	$80FA	- Print copyright message, **(Batteries Included Vx.x (C) 1984).**
33021	$80F0	- Switch to 40 column display